Multi-grid operation of WW3

Hendrik Tolman

The WAVEWATCH III Team + friends
Marine Modeling and Analysis Branch
NOAA / NWS / NCEP / EMC

NCEP.list.WAVEWATCH@NOAA.gov
NCEP.list.waves@NOAA.gov

Atmospheric and Oceanic Science

Covered in this lecture:

- Mosaic models (*ww3_multi*) with traditional regular grids.
  - Background
  - Running *ww3_multi*.
  - Hints and pitfalls.

- Background on numerics.

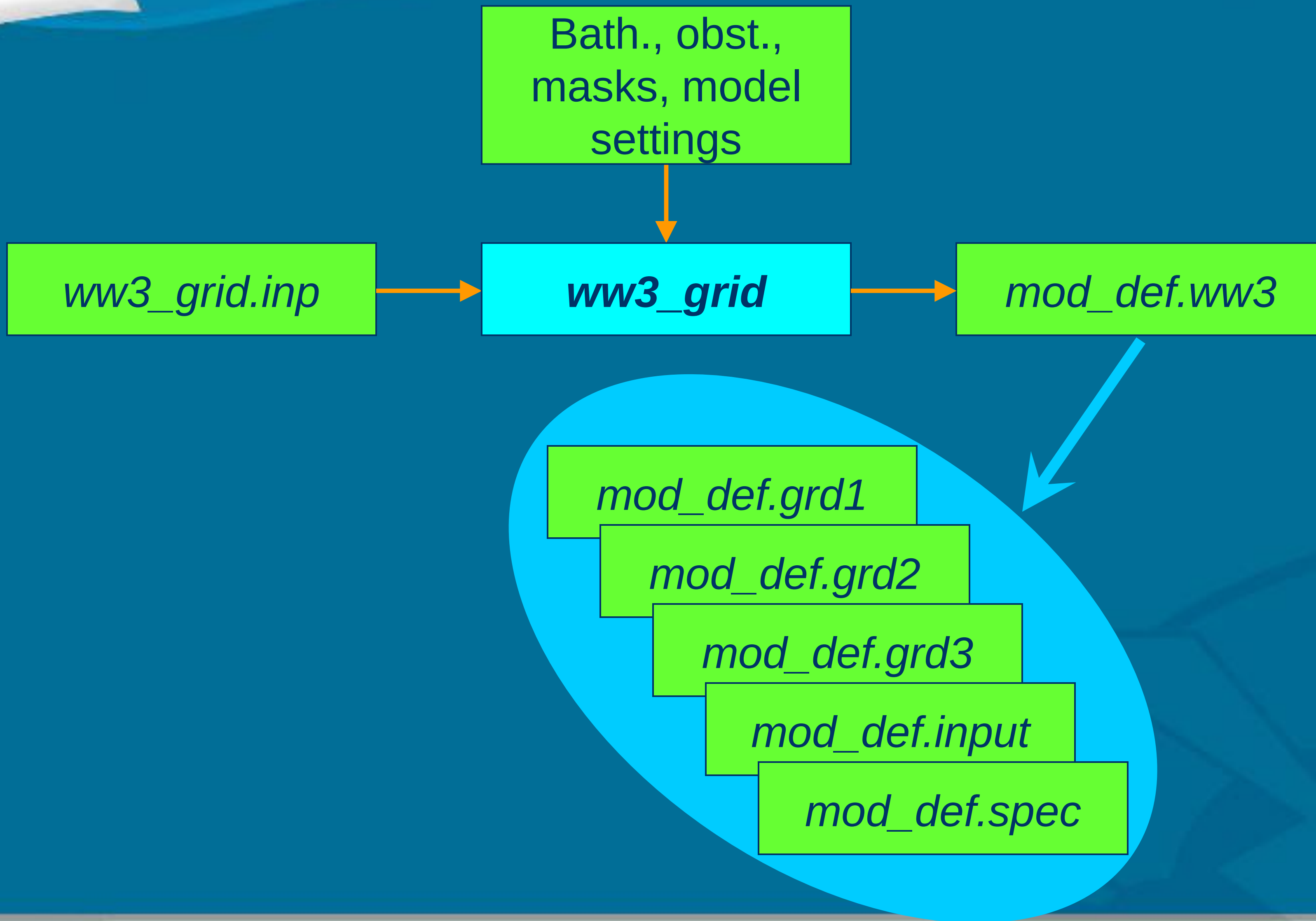
Multi-grid or mosaic model capability was introduced in model version 3.14

- Full two-way interaction between grids with variable resolution makes a mosaic of grids into a single wave model.
- Here we will concentrate on details of running the model (***ww3_multi***) only.
- Key for successful mosaic is consistency between grids.
  - Automated grid generation recommended.
  - Gridgen lecture and tutorial later today.

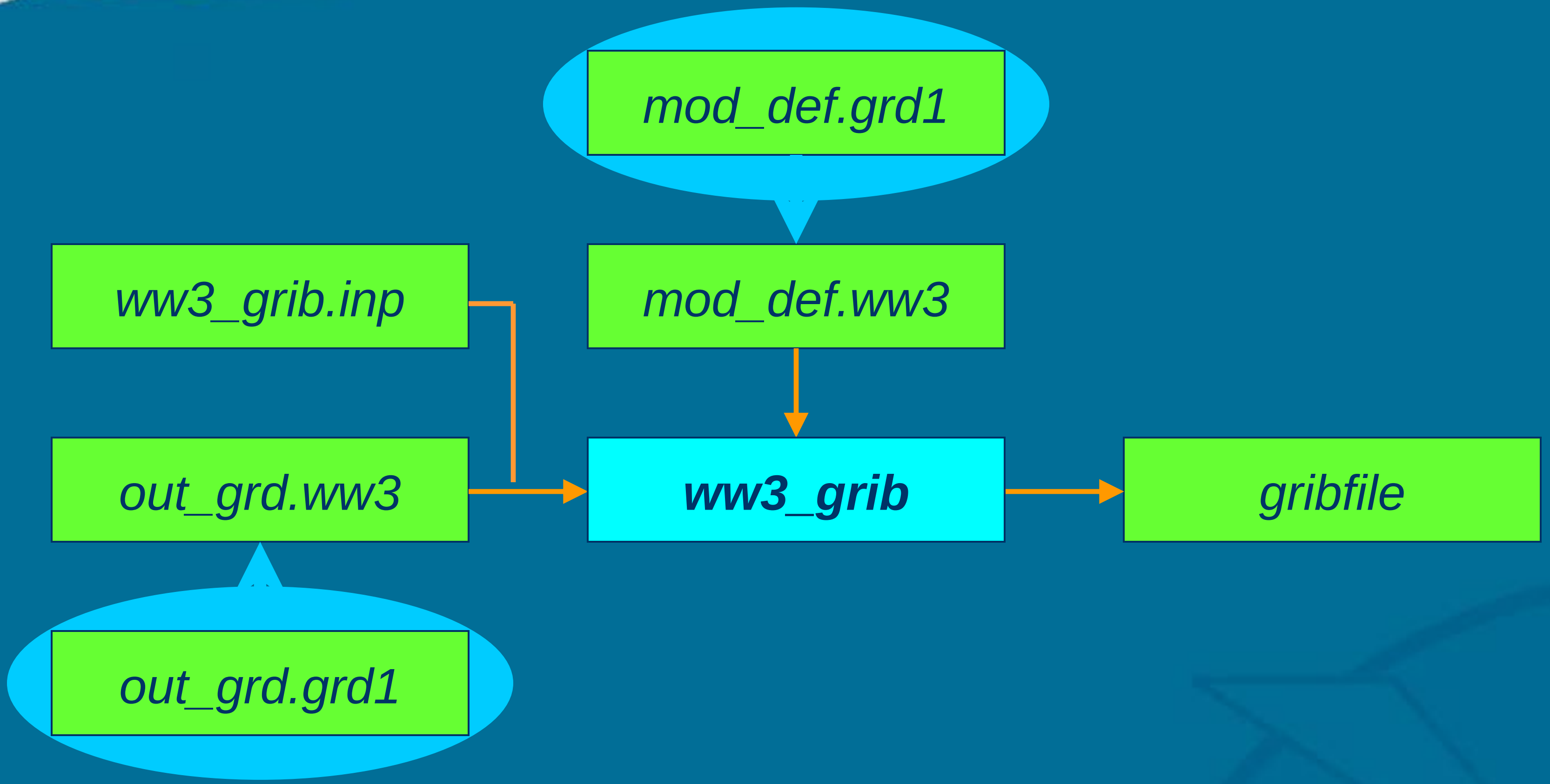
ww3_multi basics:

- ***ww3_multi*** replaces ***ww3_shel*** only, all other programs are used as before:
  - Each individual grid or data for each individual grid is processed as in the single grid model version, however,
  - ***ww3_multi*** identifies each grid by a unique file extension replacing "***.ww3***" in traditional ***ww3_shel*** model applications.
    - Copy preprocessed files to unique file name after processing, e.g., ***mod_def.ww3*** becomes ***mod_def.grd1***.
    - The opposite copy or link is needed to post-process raw model output data.
  - See following examples …..

Running *ww3_multi*



Running *ww3_multi*



ww3_multi basics cont’ed:

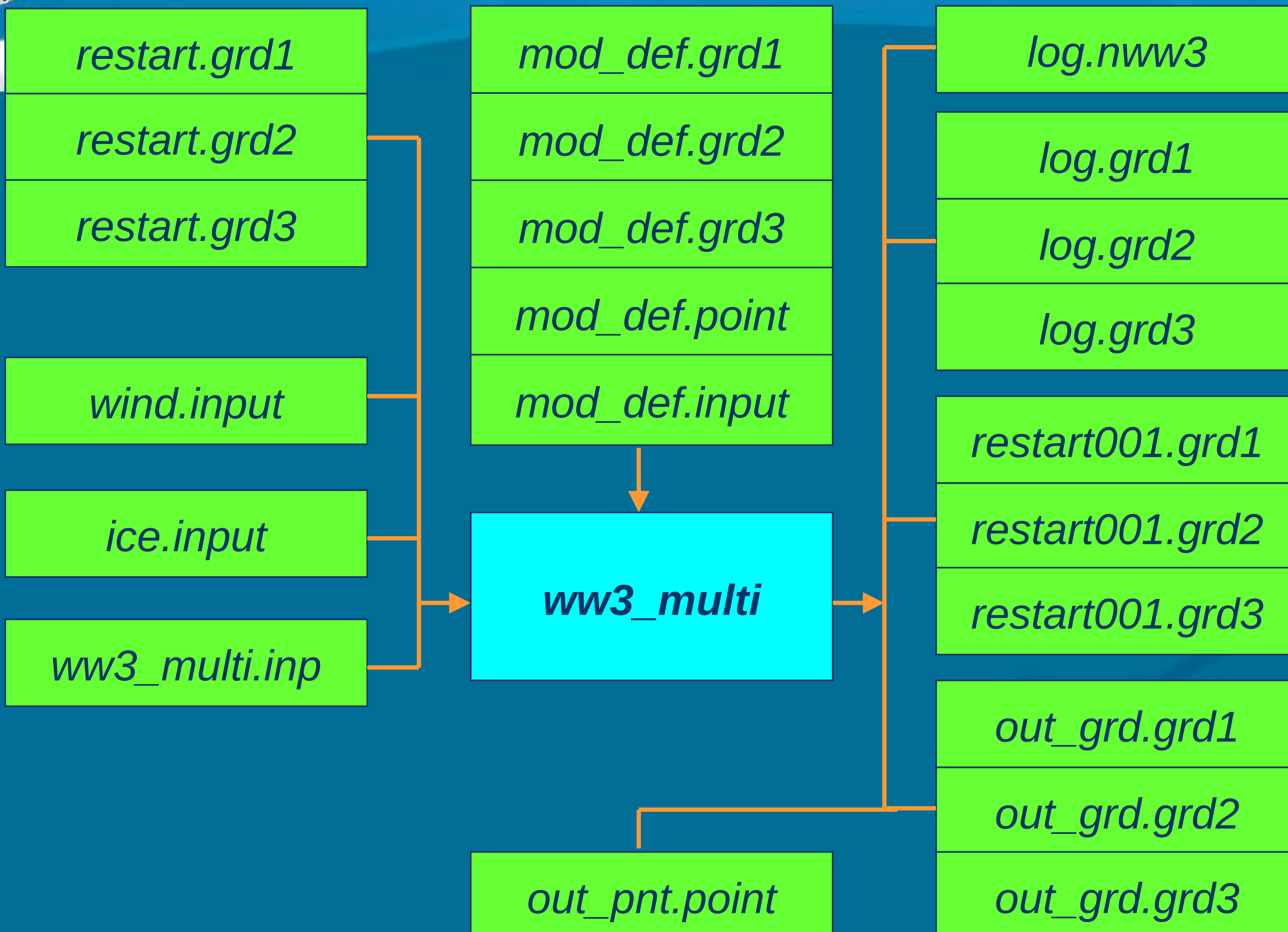
- ***ww3_multi*** recognizes three types of grids:
  - Regular model grids identifying grids that make up the mosaic.
  - Separate grids on which model input is defined.
    - Grids for the actual mosaic can get their input as preprocessed on their own grids (***mod_def.grd1*** using e.g., ***wind.grd1***), or from separate input grids with internal interpolation/averaging within the model. (optional)
  - Point output can be produced per grid, or as a single unified file. ***ww3_multi*** does not require the spectral grid to be identical throughout (unlike ***ww3_shel***), requiring an extra grid definition. (optional)

ww3_multi basics cont'ed:

- There are many ways to run ***ww3_multi***, here an example will be given of the data flow for the following model set up:
  - Three grids in the mosaic, identified with ***.grd1***, ***.grd2***, and ***.grd3***.
  - All grids are driven by ice and wind from a common grid ***.input***.
  - All grids create their own ***restart*** and ***out_grd*** files, but all point output is written to a single file (***.point***).

- See manual and test cases on how the details of the setup of ***ww3_multi*** work.
- Both idealized and real-work test cases are provided and can be used as templates.

Running *ww3_multi*



ww3_multi basics cont'ed:

- The mosaic model needs to know the relationships between grids. For this, a grid rank number is introduced.
  - Grids with (near-)identical resolution get identical ranks.
    - Grids are reconciled in overlap region.
  - Grids with lower resolution get lower rank.
    - Two-way nesting between grids.

- Grids also need a group number. Grids in the same group can be computed side-by-side in a parallel environment.

ww3_multi basics cont'ed:

- For each grid in the mosaic, the following information is needed (all set in ***ww3_grid.inp***).
  - A full grid with proper computational mask.
  - Input boundary points at which the grid expects to get data from lower ranked grids.
  - Grid time stepping information.
- Additional information provided per grid in ***ww3_multi.inp***:
  - Grid rank and group number.
  - Sources of input for grids.
  - Output per grids (or unified point output).
  - Load balancing information.
  - Output processor information if so required.

ww3_multi nesting notes:

- The lowest ranked grids in the mosaic can receive external boundary data from the file ***nest.modID***.
- The same is true for other grids, but the data has to come either from file or from the mosaic, a mix is not allowed.
- There is an option in ***ww3_multi.inp*** to dump a nesting file ***nest.modID*** for individual grids, containing all nesting data used in the mosaic.
  - This can be handy for later re-running of individual parts of the mosaic.
  - Note that the name of the **input** rather than **output** nesting file is used here.
  - By default, internal nesting data is never saved to file.
- Each grid in the mosaic can produce traditional nesting files ***nestNNN.modID*** as in ***ww3_shel***.

ww3_multi hints and pitfalls

- Build grids up from low resolution to high resolution.
  - Consistency between input points for higher ranked grids and grid masks for lower ranked grids can be tricky.
  - Automated grid generation recommended (see gridgen lecture and tutorial later today).

- Properly designed grids can be “plug-and-play”, if
  - Grids are added at consistent higher ranks, and
  - overlapping grids are introduced simultaneously.
  - Otherwise, adding or removing individual grids may result in incompatibilities between grids in mosaic.

hints and pitfalls cont’ed

- Time steps can be introduced independently per grid, however, for more transparent model (profiling) behavior, it is recommended that
  - Grids with identical rank have identical overall time steps.
  - Time steps for different grid ranks are integer multiples.
  - Note that neither are requirements.
- On parallel (MPI) file systems, first build the mosaic with all grid using all processors. Optimize later by assigning individual grids with same rank to sub-sets of processors, and by designating I/O processors.

hints and pitfalls cont'ed

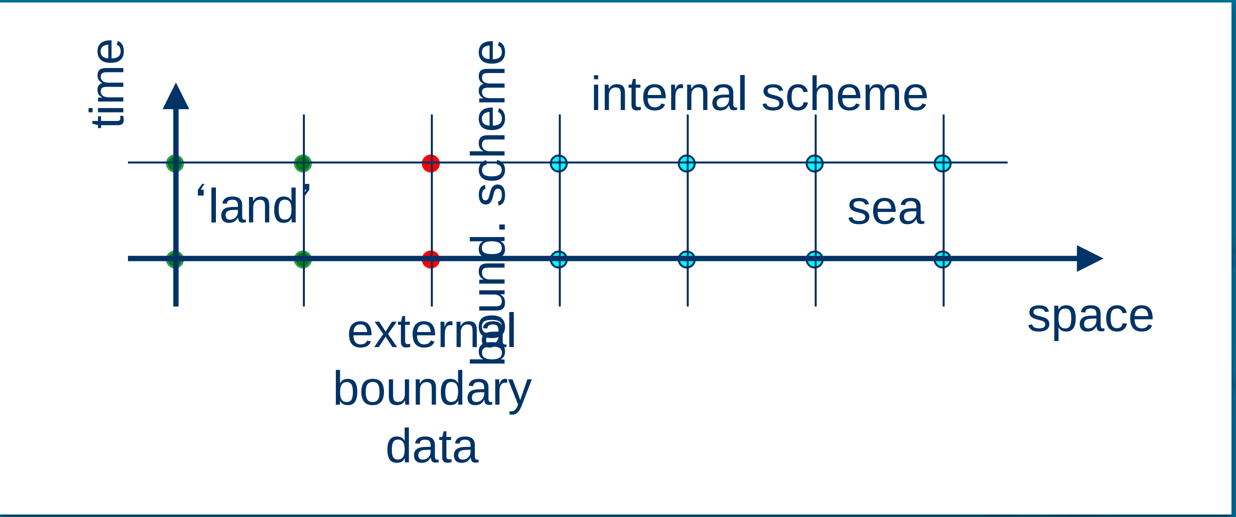
- The test ***mww3_test_01*** through ***mww3_test_05*** present a wide range of different uses of ***ww3_multi*** using idealized test cases. Some interesting cases are:
  - Providing lateral boundary data for a 2-D model by nesting it in a 1-D model with identical down-wave resolution.
  - Two groups of three grids with identical rank, with two-way nesting between both groups.
- The test cases ***mww3_case_01*** and following represent real world test case that should be suitable as a blueprint for other real-world applications.

Needed in Mosaic approach:

- Traditional data flow from low-resolution grid to high-resolution grid (traditional one-way nesting).
- Data flow from high-resolution grid to low-resolution grid (expansion to two-way nesting).
- Data exchange between grids with similar resolution.
  - Domain decomposition with capability of change of resolution between grids.

Provide data from lower ranked grid(s) at the edge of the computational domain, and apply a first order scheme locally.

- Internal boundary conditions with lower order accuracy.
- No loss of accuracy in first order scheme.
- Traditional one-way nesting approach.



In the multi-grid model the flow of boundary data should be internal, not using files.

- Updating the appropriate boundary data inside the wave model before the wave model routine is called.
- This leaves the original model unchanged, and allows for mixing of internal data flow and reading from file (choose one).
- Capability to write internal boundary data to file is useful for testing / development.

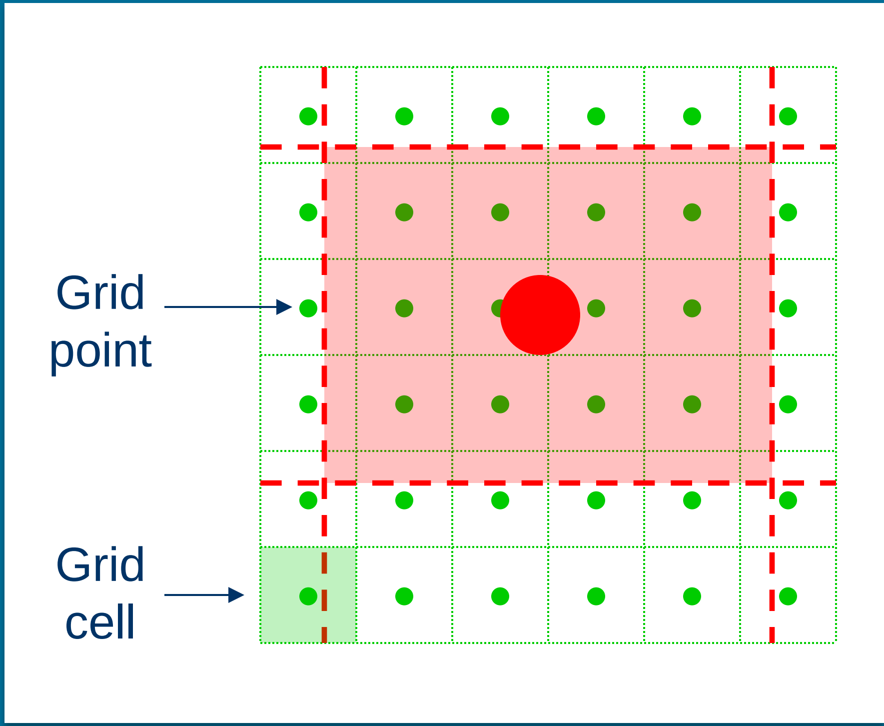
Cont' ed.

- Additional features:
  - User provides location of input boundary points in each grid. After this all data transfer is automated.
  - Each grid can receive data from each lower ranked grid simultaneously.
  - Each grid can have its own spectral resolution (within reason).

Works just like old model, but much more flexibility!

The data flow from higher ranked grids to lower ranked grid is not commonly considered in wave models.

Simple technique used here:

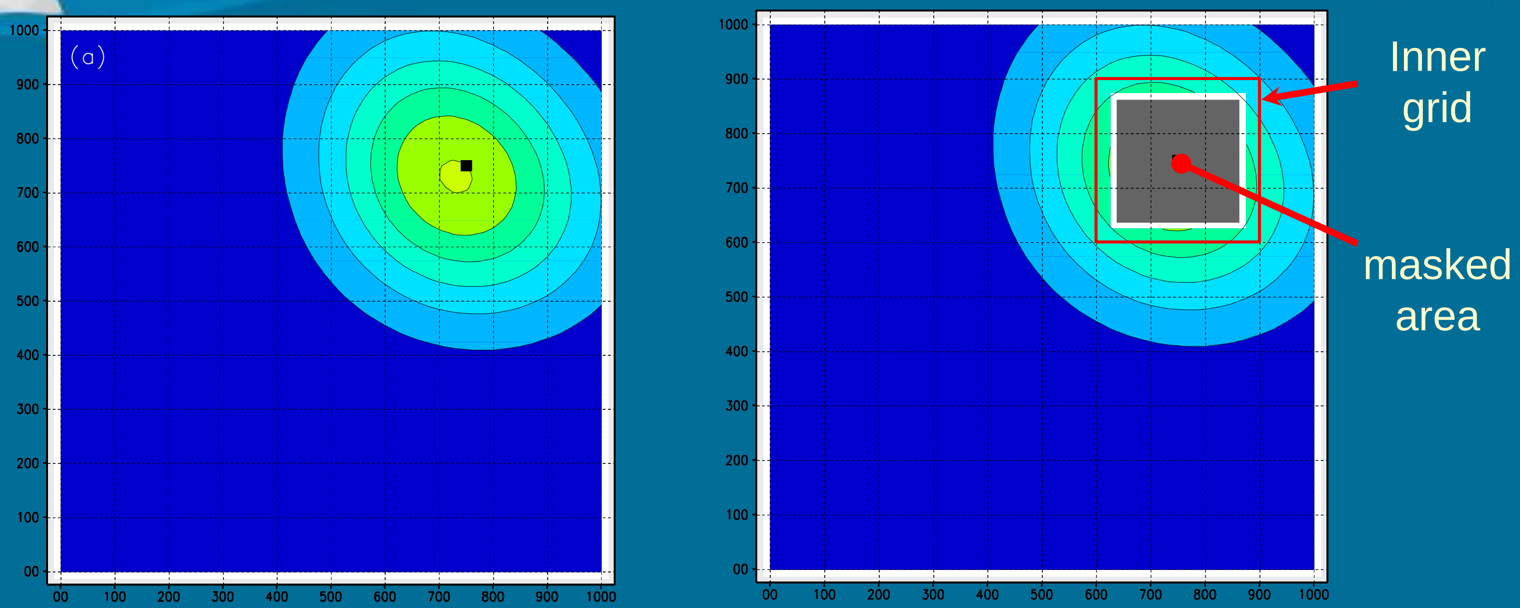


- Averaging with weight based on of high-res cell surface covering low-res cell.
- Obstructions not specially treated.

Additional considerations:

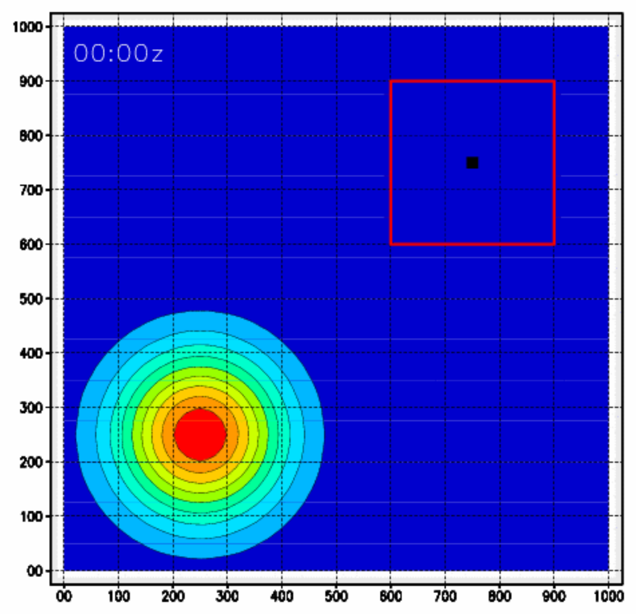
- Data from high resolution grid updated each time step of low resolution grid. This implies that only data near the boundary of the high resolution grid is ever used in the computations for the low resolution grid, depending on the stencil width of numerical scheme.
  - Option included to mask out computations in lower ranked grid.
  - Option included to mask out output similarly.

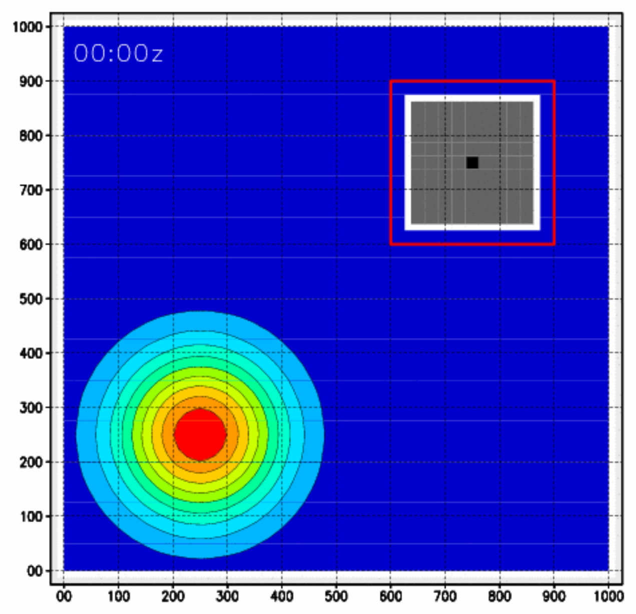
Mosaic basics



Propagation test with the first order propagation scheme and an inner grid with identical resolution. Left, full solution, or solution for single grid. Right `outer’ grid with two-way nesting and masked area.

Mosaic basics





Corresponding movie loops of full solution on the left, and masked inner grid on the right.

For overlapping grids with similar or identical resolutions it is not possible to define an order of computation, in which boundary and averaging techniques can be used.

Instead, computation is performed for all individual grids, after which the grids are ‘reconciled’ in the areas where they overlap.

Another alternative would be to consider the solutions simultaneously, but this in effect results in the generation of a single grid.

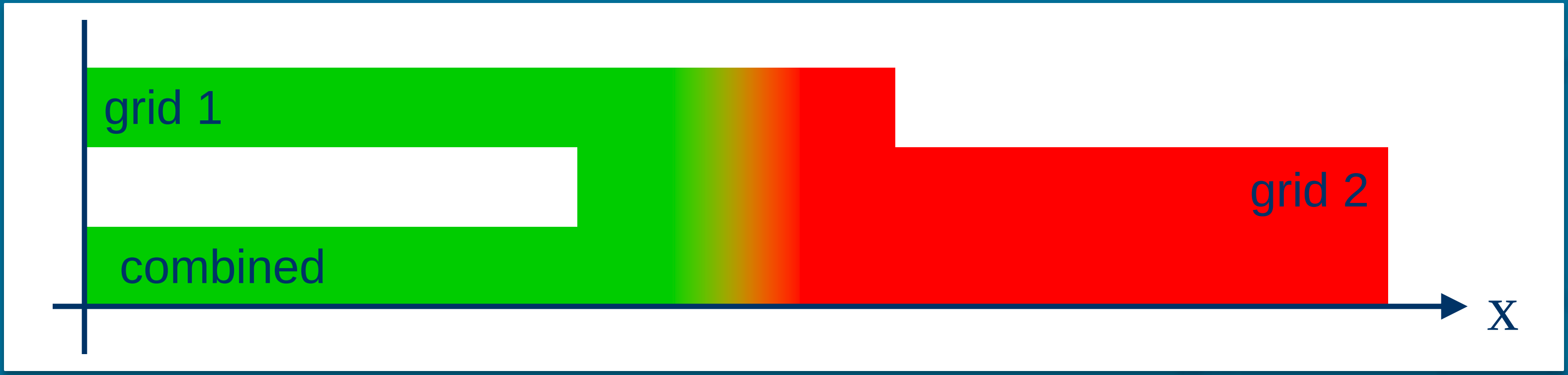
Requirements for a reconciliation technique:

- If grids are of identical resolution and grid points coincide, results should be identical to those for the corresponding single grid.
  - Internal boundary degeneration not accepted (explicit FD schemes).
  - More stringent requirement than for nests.
- The system should be sufficiently flexible to allow for (minor) differences in resolution, and / or non-coinciding grid points.

Requirement are satisfied by:

- Updates per time step taking into account depth of penetration of data at boundaries.
- Update boundary as needed.
- Spatial interpolation as needed.

Example with 1-D propagation only.

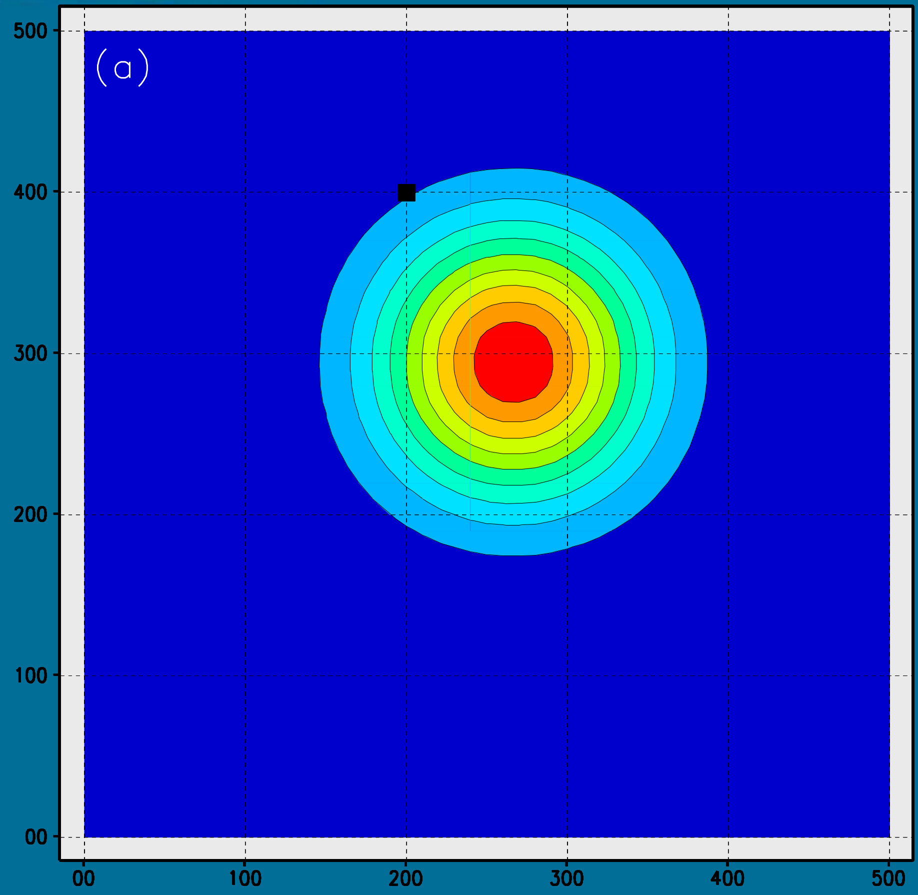


Test case:

- Swell propagation in constant water depth with single grid or three partial grids.
- Optionally coastal shallow area nested with higher resolution and concentric depth contours.
  - Offshore boundary points always needed.
  - Lateral boundary points optional.

Results for composite grids plotted consecutively in graphics.

Mosaic basics

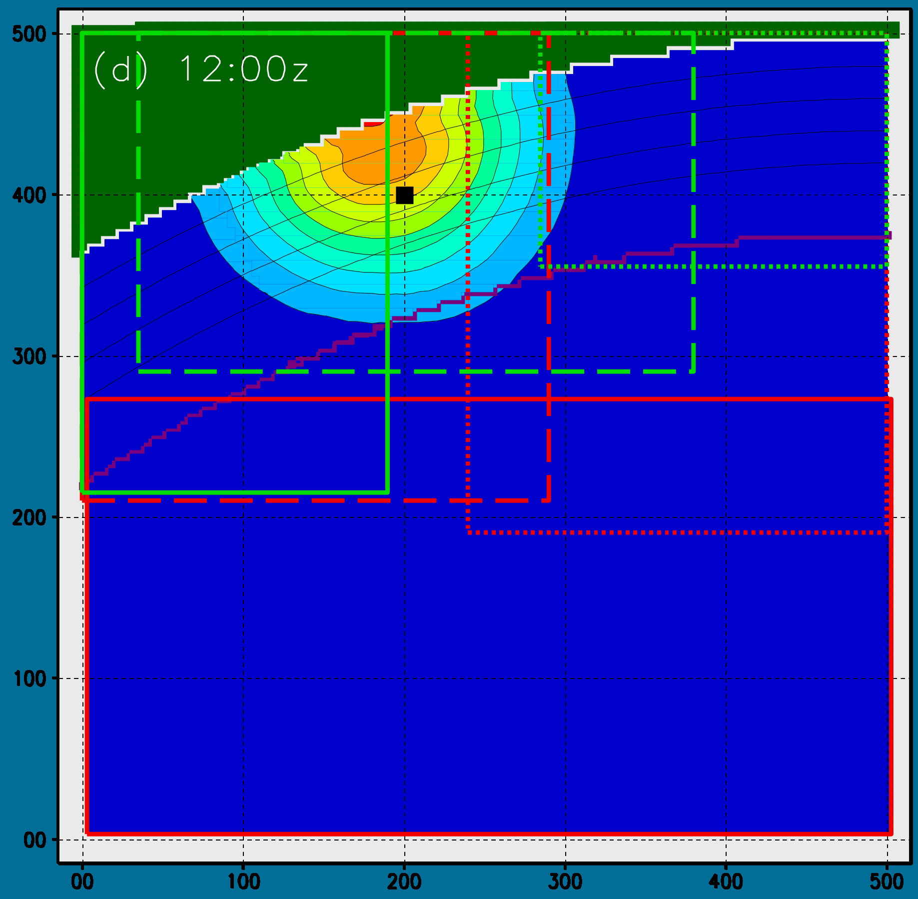


Swell propagation in low resolution grids. Results after 6h propagation.

First order scheme

Third order scheme

Mosaic basics



Swell propagation in low and high resolution grids.

Consistent grids

Inconsistent grids

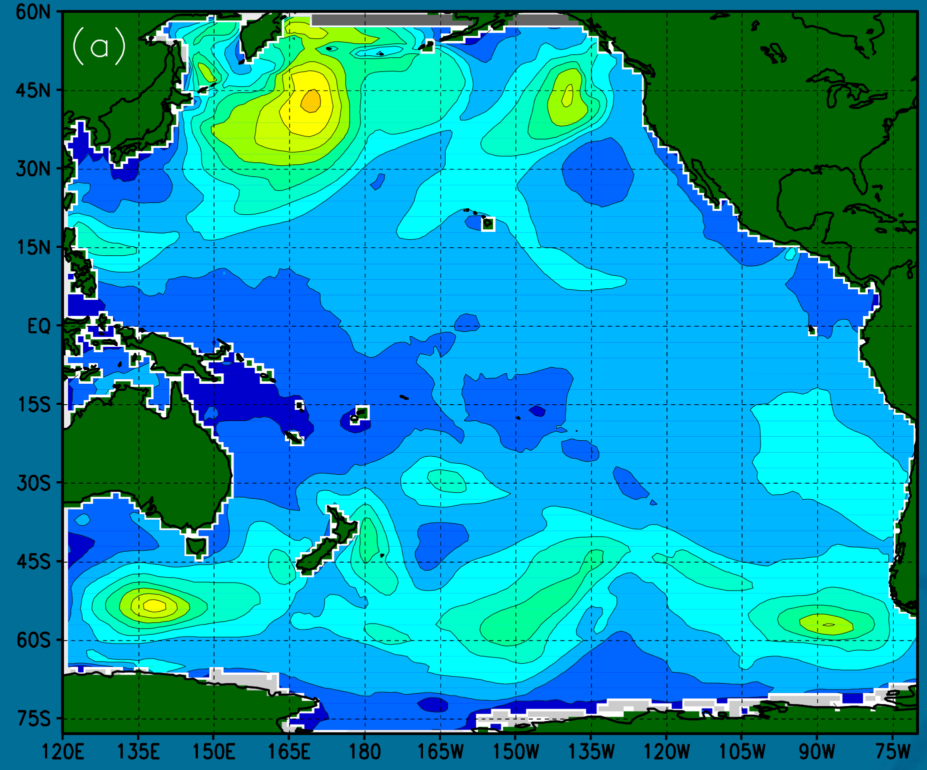
Reconciliation gives consistent results between grids.

- Identical if grids coincide in overlap.
  - Not possible for all numerical approaches,
- Small grid inconsistencies acceptable.

No special considerations for obstruction grid.

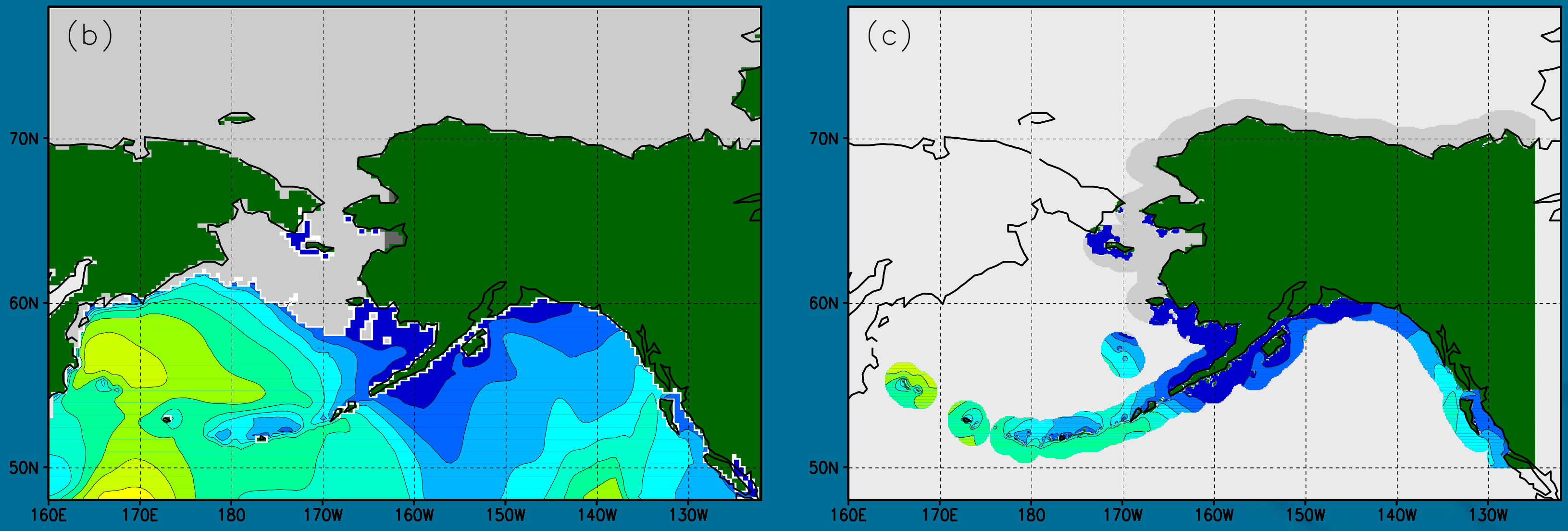
- Automated consistent obstruction generation advisable.

Modeling waves for Alaska with high coastal resolution. Start with 1° basin grid.



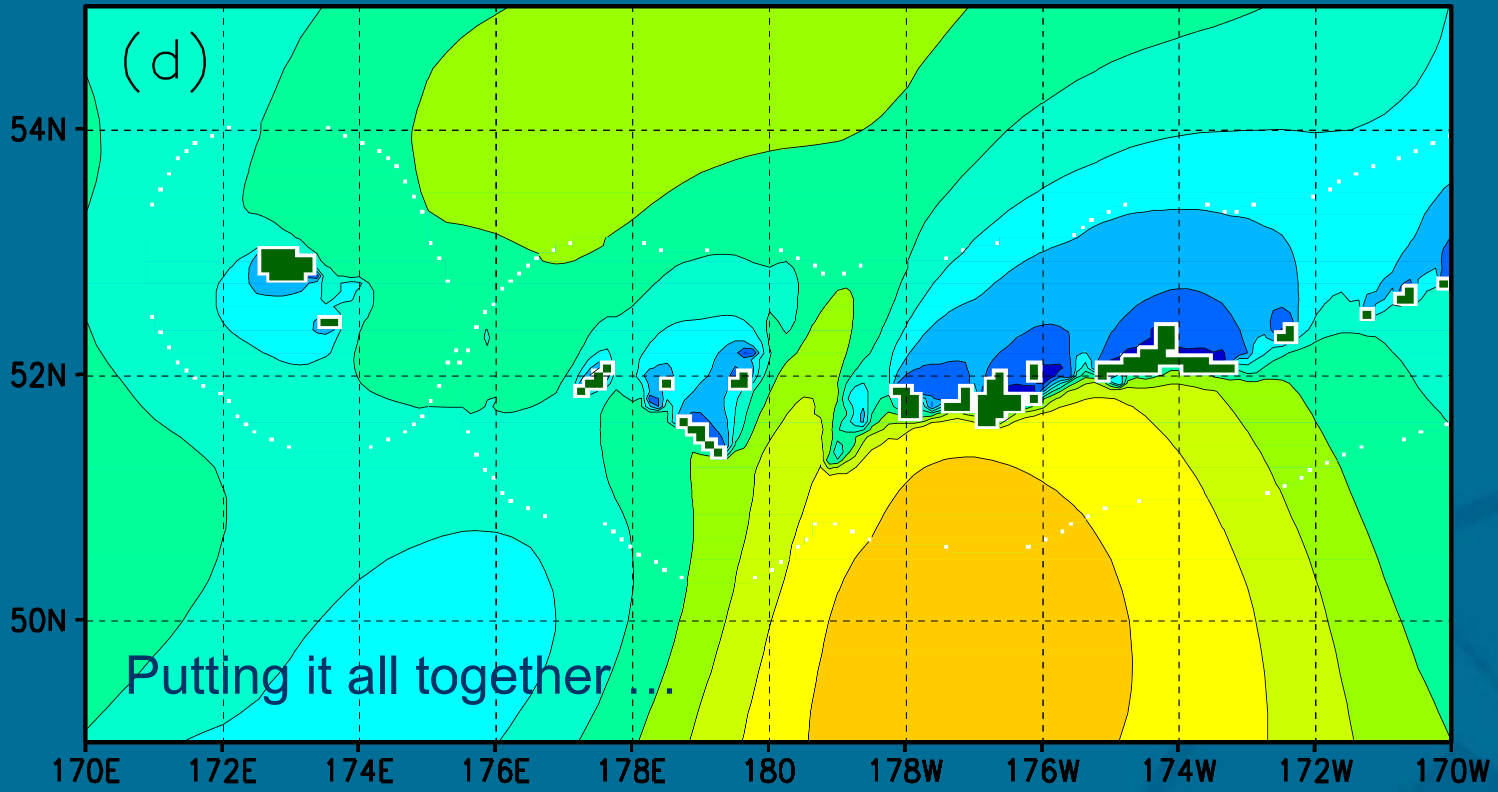
Example

Regional and coastal grids with 0.5° × 0.25° and 1/8° × 1/16° resolution, using extensive masking in the coastal grid to optimize the number of grid points.



Example

Island blocking is mostly modeled directly in the coastal grid! Additional strength of two-way nesting.



The end

End of lecture